grinders design, computer aided design,
work simulation of multidisc grinders

Józef FLIZIKOWSKI[1]
Adam MROZIŃSKI[2]
Marek MACKO[3]

COMPUTER AIDED DESIGN OF MULTI-DISC GRINDERS

Procedures of computer aided multi-disc grinders design were programmed in JAVA language and after applying them to the cell phone, they were used for the anticipation by means of genetic algorithms. The constructional solutions obtained are of logical and innovation-inspiring character.

KOMPUTEROWE WSPOMAGANIE PROJEKTOWANIA ROZDRABNIACZY WIELOTARCZOWYCH

Procedura komputerowego wspomagania konstrukcji rozdrabniaczy wielotarczowych, została napisana w języku JAVA i po zaaplikowaniu do telefonu komórkowego, gdzie została wykorzystana głownie do koncypowania innowacyjnych rozwiązań konstrukcyjnych, obsługiwała algorytmy genetyczne sztucznej inteligencji (GAAI). Uzyskane rozwiązania innowacyjne konstrukcji wielotarczowych zespołów rozdrabniających są logiczne i mają inspirujący dla twórcy charakter.

1. INTRODUCTION

IE_TEST-07_BIO software, which aims at intelligent aided multi-disc grinders design, developed in *Java*, has been implemented to mobile equipment, including cell phones, which also operate in this language of programming, in *Micro Edition* environment [3].

The cell phone with TELE-TEST-10GRAN software [3] uses genetic algorithms [1] as well as the idea of IE-TEST-07BIO procedure for intelligent aided cereal grain multi-disc design anticipation [2]. The mobile phone user has the possibility to enter data concerning:
- the material (mechanical, strength and thermal features),
- the machine (constructional features, drive method),
- the process (analysis time, neutral movement resistance, machine cutting, dynamic properties, and applicability),

[1] Uniwersytet Technologiczno-Przyrodniczy w Bydgoszczy, Wydział Inżynierii Mechanicznej; 85-789 Bydgoszcz; Al. Prof. S. Kaliskiego 7. Tel.: 52 3408293, Fax: 52 3408255, e-mail: fliz@utp.edu.pl
[2] Uniwersytet Technologiczno-Przyrodniczy w Bydgoszczy, Wydział Inżynierii Mechanicznej; 85-789 Bydgoszcz; Al. Prof. S. Kaliskiego 7. Tel./fax: 52 3408453, e-mail: adammroz@utp.edu.pl
[3] Uniwersytet Kazimierza Wielkiego, Wydział Matematyki, Fizyki i Techniki, ul. K. Chodkiewicza 30; 85-064 Bydgoszcz. Tel.: 52 3401978, Fax: 52 3414773, e-mail: mackomar@ukw.edu.pl

- the genetic algorithm parameters (population, generations number, adjustment number, genetic anticipation direction, calculation quality).

After entering the data the calculations of the objective function are performed for the selected number of population and genetic algorithm generations within the range: initial concep., the process simulation, and grinding engineering as well as the selection of the optimal solution. As a result, a configuration of effective design of cereal grain multi-disc grinder working set was obtained as well as visualization of resistance sections and functional features of the grinder.

2. MATHEMATICAL MODEL OF ANTICIPATION OBJECT

Intentionally minimized grinding indicators (coefficient, index) include: specific surface, gross energy, and the product digestibility increase, moreover, the engine and driving gear performance improvement. The intentionally minimized ones include: quasi shear loading and cross-section, motion resistance, velocity and unit energy consumption. Grinding effectiveness coefficient, as being the one that combines most of the above mentioned variables, was the grinder design optimization criterion.
Momentary integration of the grinding cross-section was used:

$$F_R = \int_{x_1}^{x_2} \left\{ b_2 + \left[R^2 - (x - a_2)^2 \right]^{1/2} \right\} dx - \int_{x_1}^{x_2} \left\{ b_1 - \left[R^2 - (x - a_1)^2 \right]^{1/2} \right\} dx \tag{1}$$

where:

a_1, a_2, b_1, b_2 - hole coordinates,
R - hole radius.

For improving the computer calculation effectiveness, geometrical dependence for a circle sector calculation was used. Calculation of the grinding cross-section F_r between two openings with center coefficients $C_1(a_1, b_1)$, $C_2(a_2, b_2)$ involves determining the distance $w = \overline{C_1 C_2}$ between the hole centers:

$$\overline{C_1 C_2} = \left[(a_2 - a_1)^2 + (b_2 - b_1)^2 \right]^{1/2} \tag{2}$$

with the assumption that the field F_r is a sum of two equal circular sectors determined by chord $B_1 B_2$, and the field of the circular sector is calculated from the formula:

$$\frac{1}{2} F_R = \frac{1}{2} (\alpha - \sin \alpha) \cdot R^2 \tag{3}$$

where:

α is a central angle $< B_1 C_2 B_2 = < B_2 C_1 B_1$.

The problem was reduced to determination of the momentary central angle. In this way the dependence (3) assumes the form:

$$F_r = (B - A) \cdot R^2 \tag{4a}$$

$$F_R = \left\{ \left(2\,\text{arctg}\,\frac{\left[1-\left(\frac{w}{2R}\right)^2\right]^{\frac{1}{2}}}{\frac{w}{2R}} \right) - \frac{w}{4R}\left[1-\left(\frac{w}{2R}\right)^2\right]^{\frac{1}{2}} \right\} R^2$$

(4b)

It is a model of grinding section for two openings in neighboring discs (levels) and satisfying the condition:

$$R < \overline{C_1 C_2} < 2R$$

(5)

Essence of solution

The aim the program, upgrading computer-aided multi-disc grinder design genetic-renewal with cell-phone (telephone aided design – TAD), was e.g. designing and execution of [3]:

- user's interface,
- menu, from the level of which it will be possible to, e.g. define the optimum criteria, configure the parameters of genetic algorithm, configure the machine parameters, process, and the material,
- subprogram storing the results of the optimization on the server or in the separate file,
- *Midlet* subprogram displaying the results of the calculations made together with a graphic representation in a form of figures.

Midlet was tested with the *J2ME Wireless Toolkit* emulator. The tool contains most of the *Midlet* application control functions, which also comply with other cell-phones. Since *J2ME Wireless Toolkit* emulator cooperates with source code editing tool and include many additional possibilities (creating files *.jad i *.jar), it was used to test the application.

Just like most applications in cell-phones, also *Midlet* contains the start image with the title of the program, list of functions available in a form of icons list in the main menu (Fig. 1 and Fig. 2).

Working in *Midlet* we use the following keys: *SELECT, BACK, OK, Menu, CLEAR and arrows UP/DOWN* on the cell–phone panel. Here is a description of respective function keys in *Midlet*:

- *SELECT* - located on the cell–phone panel; it is used to select the function from the pop-up menu,
- *BACK* - program key; it is used to come back from the current image to the previous item list of the functions,
- *OK* - program key; it is used to accept the data entered in the data fields and to accept the selections of appropriate function from the item list,
- *Menu* - program key; it is used to open the pop-up menu with a possibility of editing functions, e.g. default settings,
- *Arrows UP/DOWN* - located on the cell-phone panel; they are used to navigate through the options in the main menu, and in the pop-up menu as a well as to navigate through the date fields,
- *CLEAR* - located on the cell-phone panel; it is used to clear the data entry field.

Genetic anticipation

The main aim of *Midlet* was computer aided multi-disc grinders design with the use of standard genetic algorithm (GA) [2]. However no ready GA packages were used. Own procedure, *Genetico,* was created which fully executes the optimization jobs in the program [3]. Having selected *START PROGRAMU (program start)*, there appears item menu with four functions to select from: **O Programie (**About program)**, Wspomaganie** (Aided)**, Dane (**Data)**, Wykres (**Graph) **-** Fig. 2. Each user can select one of the options:

Fig. 1. Cell-phone start image with title of the progam computer (telephone) aided multi-disc grinders design CAD=TAD-IE

Fig. 2. Cell-phone item menu CAD=TAD-IE [3]

- information about the program **<O programie>**,
- possible configuration of data connected with the genetic algorithm and calculation process activation **<Wspomaganie>**,
- the option allows for selecting one of the three options of grinding engineering variables (Material, Machine, and Process) and for entering data in the fields **<Dane>**,
- The pop-up menu options allow for selecting default settings,
- presentation of numerical values with the calculation results **<Wyniki>** (results),
- the option includes submenu <*Pokaż>* (view) <**Wykres>** which, when selected, shows the graph presenting data entered by the user in the option <'Dane'>; it is possible to present a few parameters available in pop-up menu,
- design features of the optimal design concept **<Rozwiązanie zadania>** (problem solution).

Midlet includes the user's interface created with application programming interface (API) – in assembler, and of high level.

In *Midlet* there were two references to assembler API; first at creating the start image which is given in Fig. 1. Below a part of source code is given which refers to the start image with the *Midlet* title:

```
public class Obraz extends Canvas {
        protected Image nowy;
        protected void paint(Graphics g){
        int szerokosc = getWidth();
        int wysokosc = getHeight();
        try{
        nowy = Image.createImage("/Hammer1.png");
        }catch (IOException ex) {};
        g.setColor(0);
        g.fillRect(0,0, szerokosc,wysokosc);
        g.drawImage(nowy,szerokosc/2,20,Graphics.TOP | Graphics.HCENTER);
        Font                                czcionka                                =
Font.getFont(Font.FACE_PROPORTIONAL,Font.STYLE_ITALIC|Font.STYLE_BOLD,
Font.SIZE_LARGE);
        g.setFont(czcionka);
        g.setColor(0xffffff);
        g.drawString("WSPOMAGANIE",35,wysokosc-80,0);
        g.drawString("KONSTRUKCJI",40,wysokosc-60,0);
        g.drawString("ROZDRABNIACZY",28,wysokosc-40,0);
        g.drawString("WIELOTARCZOWYCH",16,wysokosc-20,0);}}
```

The second reference to assembler API was made at creating graphs being a graphic representation of the data entered in the option 'Dane' (Fig. 3).

The method is called with one argument which is an object of another class of assembler API – *Graphics*. The class provides methods which allow for drawing lines, rectangles, arcs, areas filled with color, as well as rendering the text on the display.

Part of source code and the result of the compilation on the emulator are given below [2, 3]:

```
public class Wykres extends Canvas{
private double[] dane_we;
private int[] dane_wykres;
private boolean gotowy;
private int wysokosc,szerokosc;
private String[] label={"sigma [mPA]","eBrutto [MJ]","dSrod [MJ]","nBio [MJ]"};
private String napis;
Wykres(){
    dane_we=new double[5];
    dane_wykres=new int[5];
    gotowy=false;
    wysokosc=getHeight();
    szerokosc=getWidth();
    napis=label[0];}
public void setLabel(int nr){
    napis=label[nr];}
public void setData(double[] tablica){
```
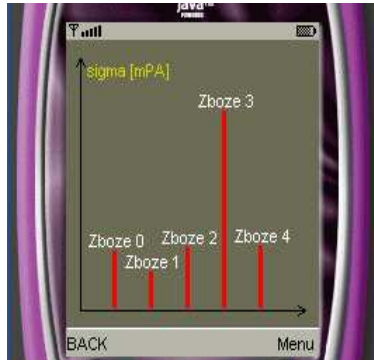
Fig. 3.   Graphic representation of the research results and TAD-IE program data [3]

Here is a set of high level API classes set in the program, together with a short description and a part of source code:

-   object *Command* - it allows for operations e.g.: *OK, BACK*, or *EXIT* in the applications:

```
private Command Com1 = new Command("START PROGRAMU", Command.SCREEN, 0);
…
public void commandAction(Command c, Displayable d) {
if (c==Com1){
   ekran.setCurrent(lista);
   lista.addCommand(Com3);
   lista.addCommand(Com4);
   lista.setCommandListener(this);
   }…
```

-   object *Form* - class Form, representing form, heritable after Screen and is used to create uses interface which consists of simpler elements, e.g. text fields, strings and labels.

```
protected Form[] formularz1;
…
formularz1 = new Form[5];
for (int i=0; i<5; i++) {
     formularz1[i] = new Form("Dane materialu: "+daneMaterial.getElement(5,i));…
```

Here is a sample source code:

-   object *ChoiceGroup* - it is one of the Form class components which allows the user to select one or more options.

```
ChoiceGroup grupa = new ChoiceGroup ("Wspol. wypelnienia przekroju psi
[1..150]",ChoiceGroup.POPUP,new String[] {" 0.01"," 0.06"," 0.12"},null);…
```

Example from *Midlet:…*
- object *TextField* - the class given on the form, allowing for displaying a single row of data entry.

formularz5.append(newTextField(fCechy[i],daneProcesTworzywo.getDeltaF(i),5,TextField.NUMERIC));...

- object *List* - just like object ChoiceGroup allows the user to select one or more options:

```
protected List lista;
String[] el = new String[] {" Dane"," Wykres"," Start Obliczenia"," Wspomaganie"," O Programie"};
Image[] obrazy = {icon1,icon2,icon3,icon4,icon5};
    lista = new List("Wybierz opcje: ", List.IMPLICIT,el,obrazy);
```

Genetic Concept Derivation

Genetic algorithm used in program TELE-TEST-10GRAN is the author's solution and consists of the following functional blocks:
- creation of the initial population,
- sorting and evaluation of the population,
- crossing the population along with sorting and evaluation.

First the initial population is created (the number of individuals assigned by the user, the main number of genes equal 11 for each individual). Genes represent the grinder features, which are used at the grinding simulation stage, the initial population, from the variability ranges for particular genes. After random drawing the population is evaluated and stored on proper boards. The next step involves crossing the population x times where x is a value established by the user. Crossing is realized by the choice of two individuals from the population and random drawing a number from the range 1 to 11. The drawn number shows us the 'cut' place of the gene and its gluing to the second "cut" part of the second individual. The new individual is introduced to the temporary board and evaluated by means of the adjustment function, e.g. the constructional solution rightness. The evaluation, i.e. the adjustment way is also stored on the temporary board. After performing all the assumed crossings, the temporary boards with the newly created individuals and the boards with their parents are combined into one big board. The board is stored and next only the best adjusted individuals are selected in a number equal to the number of individuals of the population (decided by the user). The number of population crossing repetitions is configured by the user (aid menu option), an increase in the crossing operation repetition number affects in direct proportion the final result quality and the calculation time and inversely proportionally the battery wear of the phone on which the program is being used.

Essence of the program

To test the operation, the author's genetic algorithm was tested with square functions calculations. With this algorithm, the test was to find the local minimum of the square function entered. The test results coincided with the analytical calculations.

Fig. 4 shows the relations among the classes of *Midlet*, and above also algorithms of more important program classes.
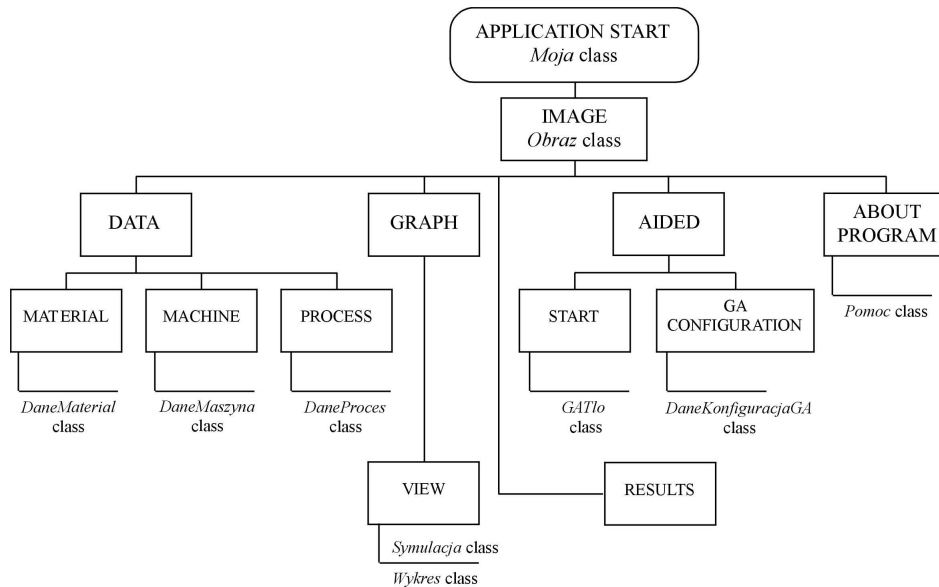
APPLICATION START
Moja class

IMAGE
Obraz class

DATA   GRAPH   AIDED   ABOUT PROGRAM

MATERIAL   MACHINE   PROCESS   START   GA CONFIGURATION

Pomoc class

DaneMaterial class   *DaneMaszyna* class   *DaneProces* class   *GATlo* class   *DaneKonfiguracjaGA* class

VIEW   RESULTS

Symulacja class

Wykres class

Fig. 4. Diagram of relations among the classes [2, 3]

Midlet installation

Midlet installation is related to the AMS (Application Management Software). This software is used to manage applications and, in general, to describe the software components. The *Wireless Toolkit* has its own implementation AMS which makes it possible to install software from two different sources:

1. From a local computer by dedicated fast connection; this mode of work is suitable especially for electronic organizers which are usually connected to the computer or laptop with which they are temporarily synchronized.
2. Via the network to which the equipment is connected, it is the most popular method of *Midlet* downloading to cell-phones and similar wireless pieces of equipment although the method can be also applied to electronic organizers connected to the network. The process of making *Midlet* sets available in the network is called *over-the-air* transmission or, in short, OTA. The AMS, making it possible to install *Midlets* from HTTP server, is connected to *Wireless Toolkit*.

Over-the-air transmission

As for OTA transmission, *Midlet* executors installed their sets on WWW servers and made their links available. The user activates the link and downloads *Midlets* to his cell-phone via WAP or Internet microbrowser. This is how to download a *Midlet* from the server to the cell-phone:

1. User browses through ACM webside.
2. User clicks on the link to download Suite.jad.
3. Server sends Suite.jad. Type MIME = text/vnd.sun.j2me.app-descriptor.
4. User sends command for *Midlet* ACM installation.

5. Server provides Zestaw.jar.

The application computer-aided multi-disc grinders design developed in *Java*, on which *Midlet* is modeled, is based on *Java Genetics Algorithms Package* (*JGAP*). It is a component of genetic algorithms written in a form of package for *Java.* The package provides genetic mechanisms which can be used easily to apply evolutionary problem-solving rules. The problem is that there are no possibilities of implementing this package to J2ME. The main limitation is that JGAP package classes are not processed. For that reason the project implements the authors solution which executes jobs of solution optimization.

Program files

Here are TAD-IE program files together with a short description of their content:

- *Moja.java* – main file of the project, inheriting data after *Midlet*; contains classes of user's interface.
- *DaneKonfiguracjaGA.java* - contains data of genetic algorithm.
- *DaneMaszyna.java* - contains data of grinder.
- *DaneMaterial.java* - contains data of material.
- *DaneMaszynaPrzyrost.java* - contains data of grinder design properties.
- *DaneProces.java* - contains data of processes.
- *DaneProcesTworzywo.java* - contains data of cereal features in grinding process.
- *Obraz.java* - contains operating assembler API class, creates an image which appears on the emulator screen at Midlet start.
- *Pomoc.java* - contains a class displaying text about the program.
- *Wykres.java* - contains a class inheriting after Canvas, drawing a graph on the display.
- *Symulacja.java* - contains a class which executes the simulation of grinder operation for selected parameters.
- *Genetico.java* - contains a class which executes genetic algorithm.
- *GATlo.java* - contains a class which creates background and "takes care of" genetic algorithm.

Optimum designsolution

The criteria of optimum design solution available are, as follows:

- minimization if value – temporary deviation of disintegration intersections from the average,
- maximization – energy efficiency.

Admissible set	LT -	LO -	LR -	PK rad/s	PR m	PLOT -	PLOR -	PPMT m	PPMR m	GPK rad/s	ST m
Competitive design	10	2	1	22,8	0,088	1	5	0,0053	0,0231	1,6	1,37

Fig. 5.   Optimum solution of 10-disc design for PVC grinding according to genetic algorithm IE_TEST-07_BIO

Below a few results obtained while working with the IE_TEST-07_BIO program are presented. They show that the solution obtained with the support of the genetic algorithm is characterized by a relatively big stability of the disintegration intersection, which was the

purpose of the defined application of this type of algorithms. A detailed analysis of effectiveness of genetic algorithms used in the design supporting process requires regular studies under the supervision of multi-disc grinder expert.
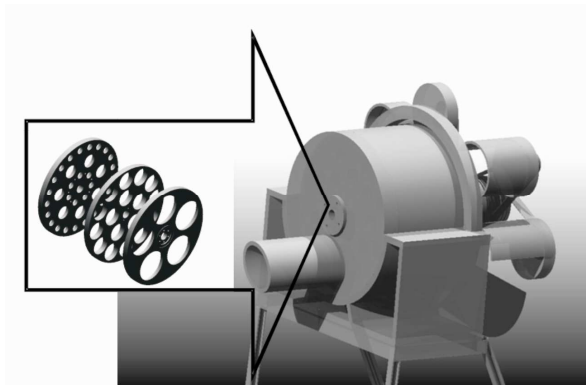

Fig. 6. Multi-disc grinder design solution in which proposed geometric features were used as a model

3. CONCLUSIONS

1. The IE_TEST-07_BIO program offers an option of the user's selection of average values calculation mode. As a result, a model was built for kinematic verification where proposed geometrical features and dependences were used (Fig. 6).
2. The average value of intersection for all the iterations in a single simulation is an additional measuring device of the solution quality for design optimization, due to a diagonal case of disintegration intersection. Therefore a potentially interesting solution should show a minimum intersection fluctuation in the "total living cycle" and its high mean value.
3. With the assumptions made, thanks to the application of the IE_TEST_07_BIO procedures applying genetic algorithms, an optimal solution was developed to design a 10-disc grinder under the conditions applied: intersection – relative force, force – relative energy, energy – relative disintegration, disintegration – relative design, material and process parameters.
4. The verification of the design solution applied was positive as far as the product quality, and process efficiency are concerned.

4. REFERENCES

[1] Goldberg D.E.: *Algorytmy genetyczne i ich zastosowania*, Warszawa, WNT 2003.
[2] Zubiedy A.Al., Flizikowski F.: Inż. i Aparat. Chem. 42, no 3, (34), 2003
[3] Network Media Group: *Projekt implementacji inteligentnego systemu (programu) wspomagania konstrukcji młynów szczególnie wielotarczowych*. Part I and II. Bydgoszcz 2003